

VŠB - Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky

## BAKALÁŘSKÁ PRÁCE

2010

Miroslav Stoniš

VŠB - Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra informatiky

Program pro stahování WWW stránek k  
offline prohlížení na platformě Pocket PC

Web Downloader for Pocket PC platform

2010

Miroslav Stoniš

## **Poděkování**

Zde bych chtěl poděkovat svému vedoucímu bakalářské práce panu Ing. Pavlu Moravcovi, Ph.D. a také panu Ing. Michalu Krumníkovi za jejich cenné rady a připomínky při vypracovávání této bakalářské práce.

## **Prohlášení**

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně.  
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne 7 května 2010

.....  
Miroslav Stoniš

## **Abstrakt**

V posledních letech se setkáváme s mohutným rozvojem mobilních zařízení, která v sobě skrývají téměř všechny funkce klasického stolního počítače. Jedním z hlavních zástupců těchto zařízení by potom byl zcela jistě kapesní počítač jinak zvaný PDA. Jednou z nepostradatelných funkcí tohoto kapesního počítače je možnost připojení k Internetu a prohlížení WWW stránek v integrovaném prohlížeči. Bohužel tvůrci PDA příliš nepočítají s použitím prohlížeče bez konektivity k Internetu. Cílem této bakalářské práce je vytvořit software, který by umožnil WWW stránky, v době kdy máme Internet k dispozici, stáhnout do paměti kapesního počítače a umožnil tak jejich prohlížení v offline režimu.

## **Klíčová slova**

PDA, stahování dat, aplikace, .Net Compact framework, Internet

## **Abstract**

In recent years there has been a massive growth of mobile devices to implement almost all of the features of classical desktop. One of the main mobile platforms is Pocket PC PDA. One of the vital functions of a handheld computer is the ability to connect to the Internet and browse web pages in the integrated browser. Unfortunately, the PDA is not suitable to browse pages without connectivity to the Internet. The aim of this thesis is to create software which would allow to download a web site into handheld computer memory, at a time when we have Internet connection available, thus allowing their viewing offline mode. The aim of this thesis is to create software that would allow web site, at a time when we have Internet available to download to your handheld computer memory, thus allowing their viewing offline mode.

## **Keywords**

PDA, data downloading, application, .Net Compact framework, Internet

## Seznam použitých symbolů a zkratek

<b>.NET</b>	– aplikační platforma společnosti Microsoft
<b>.NET Compact Framework</b>	– aplikační platforma určená pro mobilní zařízení
<b>C#</b>	– programovací jazyk vyvinutý společností Microsoft
<b>CSS</b>	– Cascading Style Sheets
<b>DHTML</b>	– Dynamic HyperText Markup Language
<b>DNS</b>	– Domain Name System
<b>HTML</b>	– HyperText Markup Language
<b>HTTP</b>	– HyperText Transfer Protocol
<b>IE</b>	– Microsoft Internet Explorer
<b>IP</b>	– Internet Protocol
<b>Java</b>	– Objektově orientovaný programovací jazyk
<b>Javascript</b>	– Objektově orientovaný skriptovací jazyk
<b>PHP</b>	– Hypertext Preprocesor
<b>PDA</b>	– Personal digital assistant
<b>URL</b>	– Uniform Ressource Locator
<b>WiFi</b>	– Wireless Fidelity
<b>WWW</b>	– World Wide Web
<b>XHTML</b>	– eXtensible HyperText Markup Language
<b>XML</b>	– eXtended Markup Language

# Obsah

<b>1. Úvod .....</b>	<b>9</b>
1.1 Struktura práce .....	9
<b>2. Použité technologie.....</b>	<b>10</b>
2.1 PDA.....	10
2.1.1 Charakteristické rysy PDA .....	10
2.2 Internet .....	10
2.3 WWW .....	11
2.4 Internetový prohlížeč .....	11
2.5 HTML .....	11
2.6 CSS .....	12
2.7 XML.....	13
<b>3. Problematika stahovače WWW stránek.....</b>	<b>14</b>
3.1 Problém stahovače .....	14
3.2 Princip stahování WWW stránek.....	14
3.2.1 Parametry stahování.....	14
3.2.2 Neočekávané chyby .....	15
3.3 Přehled dostupných stahovačů.....	16
3.3.1 HTTrack Website Copier .....	16
3.3.2 WebSite eXtractor.....	17
3.3.3 Teleport Pro .....	18
3.3.4 Offline explorer.....	19
<b>4. Implementace stahovače.....</b>	<b>21</b>
4.1 Rozbor jednotlivých tříd .....	21
4.1.1 Program.....	21
4.1.2 WebDownloader .....	22
4.1.3 IO .....	23
4.1.4 ReferenceColections .....	23
4.1.5 Settings.....	23
4.1.6 HTMLLinkExtractor.....	24
4.1.7 CSSLinkExtractor .....	25
4.1.8 Reference .....	25
4.1.9 RefDiagnostic .....	26
4.1.10 TextNorm.....	26
4.1.11 Options .....	27
4.1.12 DownloadItem.....	27
4.1.13 DownloadColections.....	27
4.1.14 OptionsManaging.....	28
4.1.15 DataManaging.....	28
4.2 Ukládání informací a nastavení .....	29
4.3 Třídní diagram .....	31
4.4 Testování aplikace .....	32

<b>5. Závěr .....</b>	<b>34</b>
<b>Literatura.....</b>	<b>35</b>
<b>A. CD-ROM.....</b>	<b>36</b>
<b>B. Uživatelská příručka: .....</b>	<b>37</b>
Postup při stahování www stránek.....	37

## Seznam obrázků

Obrázek 1:Zápis HTML kódu.....	12
Obrázek 2:Zápis CSS kódu.....	12
Obrázek 3:Princip stahování WWW stránek .....	15
Obrázek 4:Rozhraní programu WinHTTrack Website Copier .....	16
Obrázek 5:Rozhraní programu WebSite eXtractor.....	17
Obrázek 6:Rozhraní programu Teleport Pro.....	18
Obrázek 7:Rozhraní programu Offline Explorer .....	19
Obrázek 12:XML soubor pro uložení stažených projektů .....	29
Obrázek 13 :XML soubor pro uložení nastavení.....	29
Obrázek 14:Třídní diagram stahovače .....	31



# 1. Úvod

Internetové připojení je v dnešní době nepostradatelnou součástí všech mobilních zařízení na našem trhu. Z důvodů mohutného rozrůstání Internetových technologií, možností, které Internet nabízí a také z důvodů nízkých nákladů na pořízení a provoz si jej může pořídit téměř každý.

Jednou z nejpoužívanějších služeb Internetu patří zcela jistě prohlížení Internetových stránek a stahování souborů z Internetové sítě. Tyto služby můžeme využívat skrze Internetový prohlížeč, který je součástí zařízení disponujícího funkcí připojení k síti Internet.

Jedním z takových zařízení je např. PDA neboli kapesní počítač, který se dnes k síti Internet připojuje převážně pomocí technologie WiFi. I přes rozsáhlé pokrytí sítě Internet se čas od času můžeme dostat do situace, kdy nám připojení k Internetu není umožněno.

Pro tyto případy vznikl software k offline prohlížení WWW stránek, který nám umožní si požadované stránky zálohovat do paměti zařízení ještě v době, kdy máme Internet k dispozici a v případě potřeby nám tak umožnil jejich prohlížení v offline režimu. Dalším důvodem, proč si tento program pořídit by bylo stažení WWW stránek pro své známé, kteří ještě Internetové připojení nemají, nebo pro uživatele s vytáčeným připojením, kdy se platí za dobu trvajícího připojení.

Cílem této bakalářské práce je tento software vytvořit pro použití na kapesních počítačích. Výsledný software bude naprogramován v jazyku C# pro platformu .Net Compact Framework tak, aby umožňoval stahovat samotný HTML kód, kaskádové styly a Javascript při zachování původního zobrazení WWW stránky na zařízení.

## 1.1 Struktura práce

Celá práce je rozdělena do pěti hlavních částí. První částí je Úvod obsahující podkapitulu Struktura práce. Druhá část slouží pro objasnění všech pojmů dále používaných v tomto textu. U každého pojmu je stručně uvedeno vysvětlení a také jeho využití nebo výskyt v praxi.

Třetí část se věnuje již samotné problematice stahovače WWW stránek. Je zde uveden úvod do této problematiky a spolu s ním jsou také objasněny důležité funkce a možnosti stahovače. Nakonec této části je uvedeno stručné seznámení s několika dostupnými stahovači na našem trhu.

V čtvrté části je rozebrán postup tvorby vlastního řešení aplikace pro stahování WWW stránek. Na tomto místě jsou stručně popsány jednotlivé třídy aplikace, jejich implementace a jsou zde uvedeny i případné přednosti či nedostatky.

V páté a zároveň poslední části jsou uvedeny osobní poznatky a zkušenosti získané v průběhu tvorby aplikace a v samotném závěru je uvedeno stručné zhodnocení vytvořené aplikace, seznámení s její funkcí a nasazením v provozu. Jsou zde také uvedeny přednosti a nedostatky celé aplikace a nakonec jsou uvedeny příklady pro budoucí možná rozšíření aplikace.

## 2. Použité technologie

### 2.1 PDA

PDA znamená v překladu osobní digitální asistent, ale běžně je znám spíše pod názvem kapesní počítač neboli Pocket PC. Jedná se o ekvivalent stolního počítače jehož hardwarové prostředky byly vmístěny do zařízení o velikosti mobilního telefonu. V mnohém jsou si mobilní telefony a kapesní počítače podobné a dnes se již také setkáváme se snahou tato dvě zařízení spojit v jedno což je také vidět na několika zařízeních prodáváných na našem trhu.

Kapesní počítače jsou charakteristické svým ovládáním, kde se téměř veškeré úkony provádějí pomocí dotykového pera, tzv. stylusu, kterým se dotýkáme příslušných ovládacích prvků zobrazených na dotykovém displeji zařízení. Tím se zredukovalo nadměrné množství tlačítek a tak se PDA mohla omezit pouze na tlačítko pro jeho zapnutí respektive vypnutí případně další pomocná tlačítka.

Původně sloužily kapesní počítače především pro organizování času a kontaktů. V současné době jsou běžné PDA schopna přehrávat videa či spouštět různé aplikace. Mezi nejčastější operační systémy provozované na kapesních počítačích patří Windows Mobile, Symbian OS a Palm OS.[1]

#### 2.1.1 Charakteristické rysy PDA

- Hmotnost se pohybuje v rozmezí 100 až 250 gramů.
- Velikost bývá nejčastěji okolo 150 x 100 x 20 mm.
- Díky dotykovému displeji se veškeré činnosti provádějí pomocí dotykového pera, které je obdobou kurzoru myši u klasického stolního počítače.
- Není vybaveno klávesnicí, ale obsahuje systém pro rozpoznávání písma, díky kterému lze zapisovat kratší poznámky.
- Mezi nejzákladnější funkce patří synchronizace s počítačem.
- Je vybaveno procesorem, pamětí RAM, vnitřní pamětí případně doplněnou o paměťovou kartu, operačním systémem, na kterém lze provozovat naprogramované aplikace.[1]

### 2.2 Internet

Pod termínem Internet se skrývá celosvětová počítačová síť sloužící pro získávání informací, komunikaci a sdílení dat. Jsou to vlastně mezi sebou propojené jednoduché počítačové sítě, které spolu komunikují prostřednictvím protokolu TCP/IP. Mezi nejpoužívanější službu Internetu patří prohlížení WWW stránek, kterým se blíže budeme věnovat níže v tomto textu. Tyto WWW stránky jsou uloženy na počítačích připojených v síti, které nazýváme servery.

Servery jsou v síti jednoznačně identifikovány pomocí své IP adresy, která je pomocí DNS přeložena na svůj alias neboli doménové jméno k zajištění lepší navigace pro návštěvníky (dále klienty) těchto stránek.

Komunikace mezi klientem a serverem probíhá nad protokolem HTTP, který je typu požadavek odpověď, kdy klient požádá o zpřístupnění určitého dokumentu uloženého na

serveru, server tento požadavek vyhodnotí a odpoví daným návratovým kódem dle výsledku vyhodnocení daného požadavku. V případě kladné odpovědi nám server umožní požadovaný dokument přečíst, v opačném případě nám prohlížeč zobrazí chybový stav.[1]

## **2.3 WWW**

WWW je označení pro celosvětovou síť dokumentů, kterou z převážné části tvoří WWW stránky. WWW stránka je dokument zobrazený v Internetovém prohlížeči, který slouží k zaznamenávání informací a jejich zpřístupnění dalším osobám. V převážné většině se nejedná o jeden dokument ale o soubor dokumentů, které na sebe vzájemně navazují prostřednictvím odkazů.

Stránky mohou být buď statické nebo dynamické. Statické stránky zobrazují pevně napsané informace a dokud nezasáhne sám autor stránek nebo k tomu oprávněná osoba stránky budou zobrazovat stále stejný obsah. Většinou si vystačí s jazykem HTML případně doplněným o kaskádové styly.

Naproti tomu stránky dynamické používají kromě klasického HTML navíc jednu z dalších technologií například jazyk PHP, který umožňuje vytvořit Internetový obchod, různá diskusní fóra a mnoho dalších druhů stránek.

V tomto textu se omezíme na jednoduché stránky napsané v jazyce HTML (XHTML) případně doplněné o PHP skripty a kaskádové styly, kterým se budeme věnovat níže.[1]

## **2.4 Internetový prohlížeč**

Internetový prohlížeč je program, pracující pod operačním systémem, který slouží k prohlížení WWW stránek. Standardně bývá součástí operačního systému a mezi hlavní zástupce by patřili Microsoft Internet Explorer, Mozilla Firefox a Opera.[1]

## **2.5 HTML**

HTML je značkovací jazyk, pomocí kterého definujeme obsah a strukturu dokumentu. Dříve byl využíván i pro definování vzhledu stránek, ale od zavedení kaskádových stylů se od toho upustilo.

Tento jazyk používá speciální řetězce zvané tagy, pomocí kterých definuje jednotlivé části stránky. Nejdůležitější z nich jsou tagy HTML, který označuje začátek a konec zdrojového kódu stránky. Dalším důležitým tagem je HEAD, který označuje hlavičku HTML dokumentu, ve které se nachází tag TITLE, kterým zadáváme titulek stránky zobrazený v horní liště webového prohlížeče. Posledním tagem, který zde zmíním je tag BODY, pomocí kterého definujeme tělo stránky neboli samotný obsah stránky.

Všechny výše zmíněné tagy jsou párové, to znamená, že se musí vždy po použití uzavřít názvem daného tagu předcházeným zpětným lomítkem. Vedle párových tagů jsou zde také tagy jednoduché, kterým je například tag BR sloužící pro zalamování řádků.

Celý HTML dokument bývá uložen v souboru s koncovkou HTML nebo HTM. V případě využití skriptovacího jazyka PHP potom mohou mít soubory koncovky PHP.[1]

```
<html>
  <head>
    <title>Titulek stranky</title>
    <!--Toto je komentar a pisi se zde poznanky
         ktere prohlizec ignoruje-->
  </head>
  <body>
    Nejaky text, ktery se bude zobrazovat na strance.
  </body>
</html>
```

Obrázek 1:Zápis HTML kódu

## 2.6 CSS

CSS neboli kaskádové styly jsou jazyk pro formátování vzhledu Internetových stránek. Zajišťují tak oddělení kódu definujícího obsah od kódu pro definici vzhledu. Spolupráce s jazykem HTML se provádí různými způsoby a to buď se styly píše přímo do hlavičky HTML dokumentu, nebo se načtou z externího souboru, což je také častější případ.

Styly lze ovšem uplatnit také na určité tagy a to přímo v HTML kódu. Celý zdrojový kód kaskádových stylů se poté uloží do souboru s koncovkou CSS.[1]

```
<style type="text/css">
#okno {
  width:200px;
  height:400px;
}
.nadpis1 {
  font: 12px "Arial Black";
  color: red;
}
</style>
```

Obrázek 2:Zápis CSS kódu

## **2.7 XML**

XML znamená ve volném překladu rozšiřitelný značkovací jazyk, který byl vyvinut a standardizován konsorciem W3C. Tento formát umožňuje uchovávat data včetně informací o jejich významu. V dnešní době se již využívá pro své výhody také v databázových technologiích.

V mé práci využívám XML právě na uchovávání informací o nastavení aplikace a také informací o stažených stránkách. Jednotlivým XML schématům použitým v této práci se budu věnovat ve čtvrté části této práce.[1]

## 3. Problematika stahovače WWW stránek

### 3.1 Problém stahovače

Stahovač neboli downloader je obecně software sloužící pro stahování dat ze vzdálených počítačů v síti Internet a jejich ukládání na pevný disk počítače. Klasický stahovač pracuje tak, že mu zadáte URL adresu stahovaného souboru a cestu pro uložení na disk a stahovač soubor stáhne do vaší složky. Pokročilejší stahovače mohou být doplněny o vyhledávání, kdy nemusíte zadávat konkrétní URL adresu, ale zadáte pouze název požadovaného souboru a vyhledávač nabídne soubory z výsledků vyhledání.

### 3.2 Princip stahování WWW stránek

Stahování WWW stránek neprobíhá zadáním všech odkazů na soubory, které chceme stáhnout, jako tomu je u běžných stahovačů souborů, ale je zapotřebí nadefinovat kritéria, podle kterých se bude následné stahování řídit.

Je také potřeba rozlišovat, zda se bude stahovat pouze část WWW stránek, která je potřebná ke stažení nebo se budou stahovat kompletní stránky ze zadané domény. Standardně se zadá stahovači odkaz na požadovanou stránku a nadefinují se parametry, podle kterých se stahování stránky řídí.

#### 3.2.1 Parametry stahování

- zadání názvu projektu
- zadání URL adresy stahované stránky
- zadání místa uložení stahovaných stránek
- hloubka procházení WWW stránek
- filtrace stahovaných souborů
- omezení dle počtu stažených souborů
- omezení dle velikosti souboru
- omezení dle celkové velikosti stažených dat
- zadání přihlašovacích údajů
- omezení dle zadaného času stahování
- omezení počtu spojení

Pro samotné stahování jsou nutné pouze první dva výše uvedené parametry, kdy poté aplikace implicitně počítá se stahováním všech souborů a rozsah stahovaných stránek je brán v rámci domény, pokud není uvedeno jinak. Nyní nám již nic nebrání ke spuštění stahování.

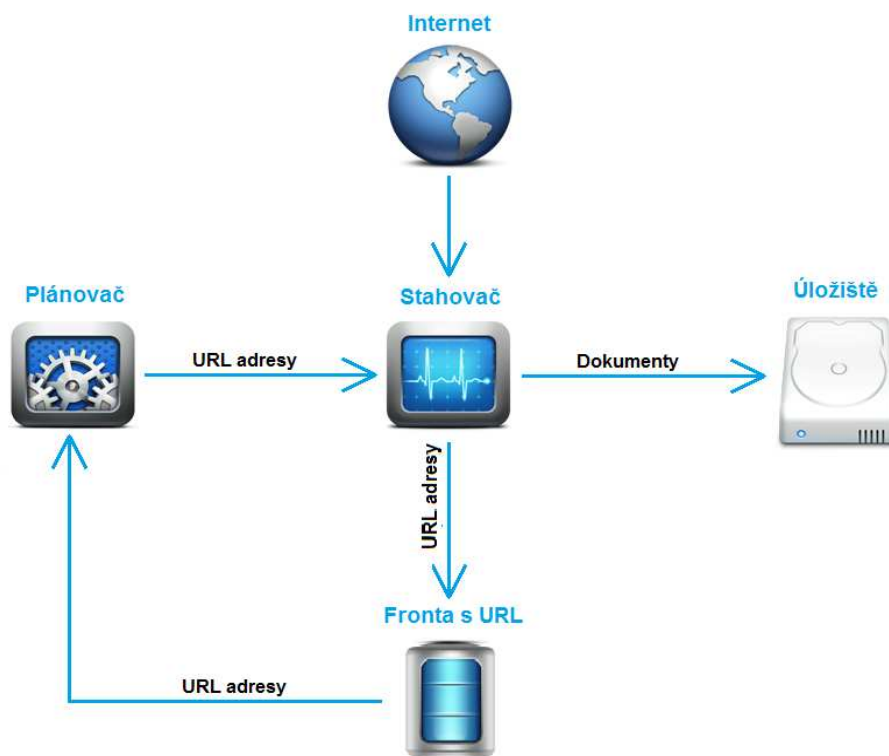
Samotné stahování pracuje na principu procházení (parsování) zdrojového kódu a současném vyhledávání odkazů. Tyto odkazy se standardně vyhledávají pomocí regulárních výrazů, což jsou výrazy, které definují tvar výskytu URL adresy ve zdrojovém kódu WWW stránky. Při nalezení URL adresy ve zdrojovém kódu se tento odkaz vyhodnotí zda odpovídá kritériím, která jsme nastavili před samotným stahováním tj. zda se jedná o odkaz ve stejné doméně jako zadaná stahovaná adresa, nebo zda odpovídá nadefinovaným typům stahovaných souborů. V případě kladného vyhodnocení se tento odkaz uloží do pracovní paměti vyhrazené

pro nalezené odkazy a procházení pokračuje ve hledání dalších odkazů dokud se nenarazí na konec souboru. V případě že odkaz není podporován je tento odkaz jednoduše přeskočen a vyhledávání pokračuje dále opět dokud nenarazí na konec souboru.

Pracovní paměť pracuje jako fronta tudíž odkazy se do této paměti uloží a jejich postupné vyjmutí se řídí podle pořadí, ve kterém byly do fronty vloženy. Tedy odkaz vložený do fronty jako první bude také první z fronty vyjmut. Procházení zdrojových kódů a ukládání odkazů pokračuje do té doby dokud se ve frontě nacházejí nějaké odkazy nebo se daří vyhledávat nové odkazy či nedojde k neočekávanému přerušení stahování z důvodu neočekávané chyby, která není ošetřena potlačením této chyby.

### 3.2.2 Neočekávané chyby

- nefunkční nebo již neexistující odkaz
- výpadek připojení k Internetu
- neoprávněný přístup k souboru
- špatný zápis odkazu ve zdrojovém kódu



Obrázek 3: Princip stahování WWW stránek

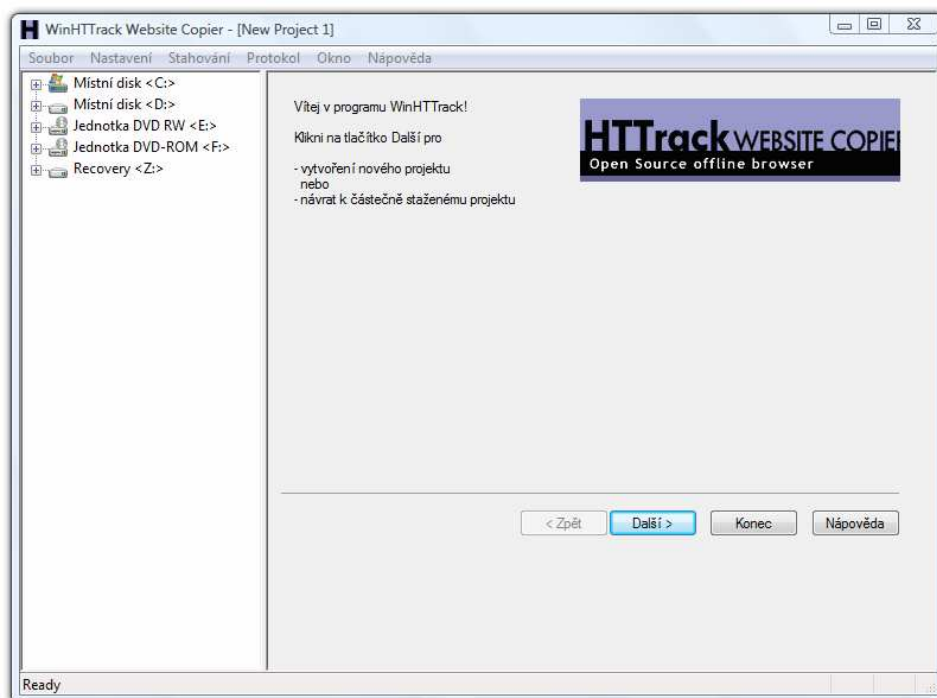
### 3.3 Přehled dostupných stahovačů

Všechny níže zmíněné aplikace jsou si svými funkcemi velmi podobné. Ve většina aplikací probíhá stahování souborů ve více vláknech, čímž se výrazně sníží doba stahování. Hlavním faktorem ovlivňujícím dobu stahování však stále zůstává rychlost připojení k Internetu, kterým je rychlost stažení limitována. Dalším faktorem bude také nastavení úrovně hloubky zanoření, kdy se většinou používá hodnota 1 nebo 2. Při vyšších hodnotách může stahování trvat delší dobu.

Také se může stát, že potřebujete stáhnout jen čisté WWW stránky pro pozdější přečtení informací a z důvodů špatného nastavení se nám mohou stahovat objemné archívy, které třeba vůbec nevyužijeme.

Proto je zapotřebí nastavit při stahování omezení pomocí kterých nastavíme parametry stahování jen pro stažení nejn nutnějších souborů. Pomocí tohoto přehledu jsem určil nejužitečnější funkce a nastavení, které jsem následně aplikoval při tvorbě vlastního software pro stahování WWW stránek.

#### 3.3.1 HTTrack Website Copier



Obrázek 4: Rozhraní programu WinHTTrack Website Copier



Jedná se dle mého názoru o nejlepší aplikaci tohoto typu a to hned z několika důvodů. Je vyvíjena jako open-source a tudíž si jí můžeme stáhnout zdarma bez jakýchkoliv omezení.

Práce v tomto programu je velmi jednoduchá a navíc je zde možnost přepnutí i do českého jazyka. Přehledně rozdělené okno má ve své levé části zobrazenou adresářovou strukturu počítače a v pravé části se nachází velmi podrobný a propracovaný průvodce, díky kterému může tuto aplikaci používat i méně zblhlý uživatel. Mezi další přednosti této aplikace patří možnost aktualizovat již jednou stažené stránky, nebo navázat na přerušené stahování stránky. Další výhodou by také mohla být možnost sledovat stahované soubory a během stahování možnost zrušení stahování objemných nebo nežádoucích souborů bez přerušení celého stahování.

Aplikace spolupracuje se všemi prohlížeči. Tento program lze nejen ovládat pomocí popisovaného grafického rozhraní, ale je zde také možnost ovládání z příkazové řádky.[2,3,4]

### **Souhrnné informace:**

Licence: Freeware

Autor: Xavier Roche

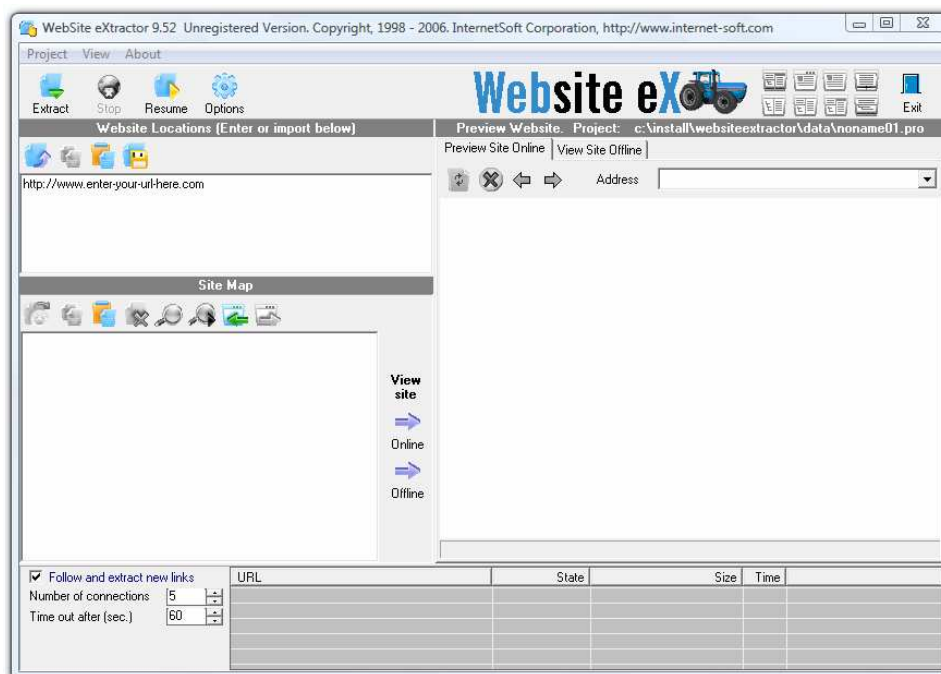
Potřeba instalace: Ne

Domovská stránka: <http://www.httrack.com>

Velikost programu: 3,7 MB

Operační systém: Microsoft Windows 95/98/ME/NT/2000/XP/Vista/7

### **3.3.2 WebSite eXtractor**



**Obrázek 5: Rozhraní programu WebSite eXtractor**

Tento program je důstojnou konkurencí výše zmíněné aplikace, ovšem jedná se o shareware aplikaci, takže pro plnohodnotnou práci bude třeba aplikaci zaregistrovat. Opět je zde možnost rozsáhlého nastavení stahování, které je však globální a to znamená, že je platné pro veškeré stahované weby na rozdíl od výše zmíněné aplikace, která měla možnost zadat nastavení pouze pro konkrétní projekt. Dalšími nevýhodami by potom bylo postrádání češtiny a zbytečně složité ovládání.[2,3,4]

## Souhrnné informace:

Licence: Shareware

Autor: Asona

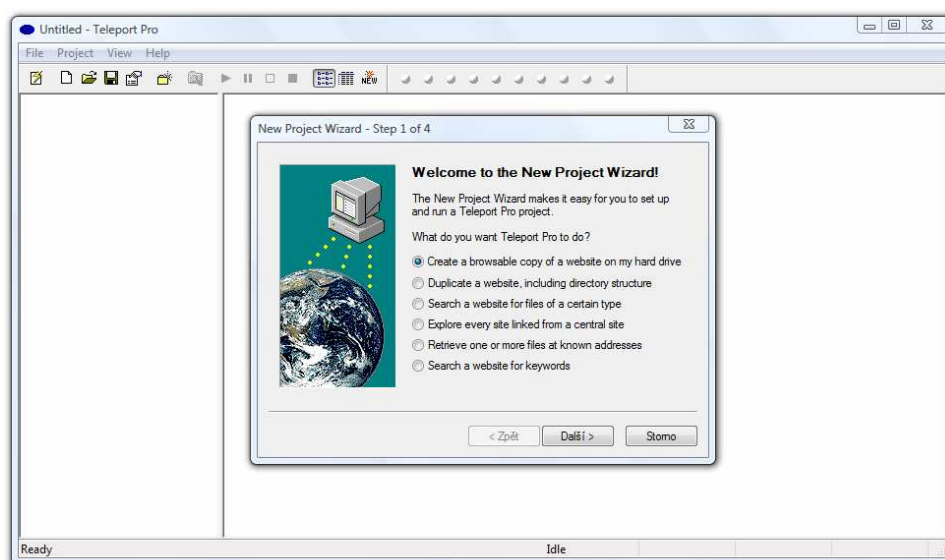
Potřeba instalace: Ano

Domovská stránka: <http://www.esalesbiz.com>

Velikost programu: 813,4 kB

Operační systém: Microsoft Windows 95/98/ME/NT/2000/XP/Vista

### 3.3.3 Teleport Pro



Obrázek 6: Rozhraní programu Teleport Pro

Jedná se o další aplikaci funkčně velmi podobnou výše zmíněným aplikacím, ovšem nese sebou omezení, kterým je při použití neregistrované verze možnost stažení pouhých 500 souborů na jeden projekt. Většinou si s tímto množstvím běžný uživatel vystačí, při větší hloubce zanoření již však toto množství stačit nebude. Na první pohled aplikace vypadá jednoduše a tento aspekt také přispívá k přehlednosti celé aplikace. Pro práci s aplikací je, jak už je zvykem, vytvořen průvodce, který nás provádí jednotlivými kroky nastavením stahování dané stránky.

Kromě samotného stažení webu je zde také možnost přesně specifikovat typy souborů, které mají být stahovány a dokonce je také možné zadat přesný název souboru, který by se měl na webu nacházet.

Další nevýhodou je také postrádání češtiny, kdy její nepřítomnost může činit začínajícím uživatelům nemalé potíže. Tato aplikace dále nabízí možnost stahování až deseti stránek současně, zpřístupňuje heslem chráněná místa, filtruje soubory a vyhledává podle klíčových slov. Aplikace umí pracovat s HTML 4.0, CSS2, DHTML a podporuje i aplety v jazyce Java.[2,3,4]

## Souhrnné informace:

Licence: Shareware

Autor: Tennyson Maxwell Information Systém, Inc.

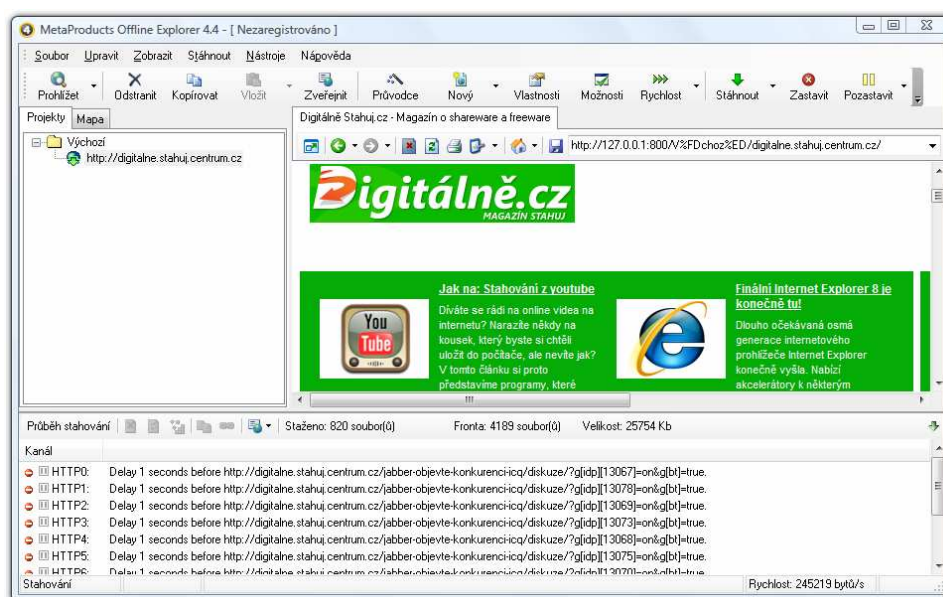
Potřeba instalace: Ano

Domovská stránka: <http://www.tenmax.com>

Velikost programu: 841,3 kB

Operační systém: Microsoft Windows 98/ME/2000/XP/Vista

### 3.3.4 Offline explorer



Obrázek 7: Rozhraní programu Offline Explorer

Tato aplikace se řadí mezi absolutní špičku ve svém oboru. Bohužel se opět jedná o placenou aplikaci takže sebou nese omezení a to konkrétně 30-denní lhůtu na vyzkoušení. Předností je opět kvalitně a přehledně uzpůsobené rozhraní s podrobným průvodcem, který uživatele doprovází ve všech krocích stažení a všechny kroky jsou navíc podrobně doplněny o vysvětlující popisky. Podstatnou výhodou zcela jistě také shledáme v podpoře češtiny. Kvalitně zpracované možnosti nastavení a vysoká rychlost stahování s podporou téměř veškerých

technologií uplatňovaných při tvorbě WWW stránek. Navíc lze uvnitř okna aplikace Vámi stažené stránky přímo prohlížet.[2,3,4]

### **Souhrnné informace:**

Licence: Shareware

Omezení: 30 dní

Autor: MetaProducts Corporation

Potřeba instalace: Ne

Domovská stránka: <http://www.metaproducts.com>

Velikost programu: 813,4 kB

Operační systém: Microsoft Windows 98/ME/NT/2000/XP/Vista

## 4. Implementace stahovače

V následujících řádcích si rozebereme vlastní implementaci stahovače a také si popíšeme funkce jednotlivých tříd. Aplikace byla programována ve vývojovém prostředí Microsoft Visual Studio 2005. Zpočátku jsem vyvíjel pro platformu .NET Framework verze 3.5 pro snadnější a rychlejší odladění zdrojových kódů.

Jakmile byla aplikace zcela funkční na stolním počítači, začal jsem pracovat na její implementaci pro platformu .NET Compact Framework verze 3.5. V podstatě stačilo vytvořit pouze uživatelské rozhraní aplikace, protože veškeré třídy a balíčky použité při vývoji pro .NET Framework verze 3.5 byly podporovány i platformou .NET Compact Framework verze 3.5.

### Kompletní stahovač tvoří tyto třídy :

- *Program*
- *WebDownloader*
- *IO*
- *HTMLinkExtractor*
- *CSSLinkExtractor*
- *Reference*
- *ReferenceCollection*
- *Settings*
- *RefDiagnostic*
- *Options*
- *OptionsManaging*
- *TextNorm*
- *DownloadItem*
- *DownloadCollections*
- *DataManaging*

### 4.1 Rozbor jednotlivých tříd

V této kapitole se seznámíme se všemi třídami, které tvoří kompletní stahovač. U každé třídy je uveden stručný popis její funkce a výpis jejích metod.

#### 4.1.1 Program

Třída *Program* plní úlohu vstupního bodu při spuštění aplikace respektive tuto úlohu plní metoda *Main*, která je v této třídě obsažena. Jediné co tato metoda provádí je to, že vytvoří instanci třídy *WebDownloader*, čímž se zobrazí na obrazovce zařízení samotné grafické uživatelské rozhraní aplikace.

### 4.1.2 WebDownloader

Třída *WebDownloader* je třídou, která definuje vzhled a ovládání rozhraní aplikace. Jelikož celá aplikace pracuje pouze s jedním oknem, proto se v této třídě nachází implementace všech grafických ovládacích prvků použitých v této aplikaci včetně všech obslužných metod pro tyto prvky.

Rozhraní aplikace je zobrazováno v jednom okně, ve kterém je na výběr z následujících čtyř karet.

#### Karta Stažení

Tvoří základní obrazovku s vstupními poli pro zadání názvu projektu, zadání URL adresy stahované stránky, tlačítka start a stop pro spuštění respektive přerušení stahování a textovým polem pro zobrazení informací o právě stahovaném souboru.

#### Karta Stažené

Slouží pouze pro zobrazení seznamu korektně stažených stránek. Zde se budou přidávat názvy projektů průběžně stahovaných WWW stránek. Tyto položky lze v případě potřeby smazat označením položky a kliknutím na tlačítko Odstranit.

#### Karta Nastavení

Umožňuje uživateli nadefinovat pravidla, kterými se stahovač řídí při stahování stránek.

#### *Seznam parametrů:*

- Úroveň zanoření – definuje hloubku procházení stránek.
- Typy stahovaných souborů – určuje typy stahovaných souborů
- Adresář s uloženými weby – umožňuje uživateli zadat cestu pro ukládání stránek

#### Karta O aplikaci

Má pouze informativní charakter a stručně popisuje funkci aplikace a uvádí Autora aplikace včetně jeho emailové adresy.

Kromě rozhraní však tato třída definuje spoustu důležitých metod. Nejdůležitější metodou této třídy je metoda *startDownloading*, která řídí celý proces stahování WWW stránek. Nejprve vyhodnotí, zda se ve frontě s dočasně uloženými URL adresami ještě nachází nějaké URL adresy. V případě že ano, vezme se první v pořadí a vyhodnotí se, zda se jedná o HTML dokument, kaskádový styl nebo jiný soubor. V případě HTML dokumentu a kaskádových stylů by následovalo procházení dokumentu s vyhledáváním dalších URL odkazů, nebo v případě jiného souboru jednoduché stažení tohoto souboru.

### **Implementované metody :**

- *loadCustomSettings* – načítá uživatelem definované nastavení
- *loadDefaultSettings* – načítá výchozí nastavení
- *initialize* – inicializuje potřebné proměnné
- *actualizeSettings* – aktualizuje uživatelem změněné nastavení
- *testInit* – testuje zda jsou zadány potřebné údaje před samotným stahováním
- *DoUPDAte* – aktualizuje informace v zobrazené v textovém poli

#### **4.1.3 IO**

Třída *IO* v sobě skrývá metody pro přístup k souborovému systému kapesního počítače a zároveň pro přístup k síti Internet. Všechny metody této třídy jsou statické, takže se pro jejich použití nevytváří instance třídy a jsou dostupné všem ostatním třídám.

### **Implementované metody :**

- *downloadContent* – vrací zdrojový kód stránky
- *downloadFile* – umožňuje jednoduché stažení souboru
- *GetStorageCardFolder* – vrací odkaz na paměťovou kartu
- *createDirPath* – vytváří adresářovou strukturu v paměti PDA
- *writeHTML* – ukládá zdrojový kód do souboru

#### **4.1.4 ReferenceCollections**

Tato třída implementuje funkci fronty, do které se dočasně ukládají URL adresy vyhledané při procházení zdrojových kódů stránek. Oproti klasické frontě se však uložené položky při načítání nevyjímají ale zůstávají ve frontě i nadále uloženy s tím že používá ukazatel na následující vyjímání prvek.

### **Implementované metody :**

- *addReference* – vkládá URL odkaz do fronty
- *getNextReference* – načte z fronty v pořadí další URL odkaz
- *isContended* – zjišťuje zda je daný URL odkaz
- *getFileName* – vrací název souboru včetně jeho přípony
- *printContent* – vypíše do konzole seznam URL odkazů uložených ve frontě

#### **4.1.5 Settings**

Třída *Settings* slouží k dočasnému uchovávání všech potřebných parametrů, které jsou pro stahování nezbytné a umožňuje tak jejich poskytnutí během běhu aplikace. Uchovávají se zde údaje jako název projektu, URL adresa stahovaných stránek, výběr stahovaných souborů, úroveň zanoření a cesta pro uložení stahovaných souborů.

Tato třída neslouží k trvalému uložení informací o nastavení, k tomuto účelu slouží třída *OptionsManaging*, která je popsána níže v tomto textu.

### Implementované metody :

- *getDir* – vrací cestu pro ukládání stažených souborů
- *getNestLevel* - vrací úroveň zanoření
- *getCapture* – vrací povolení či zakázání zachytávání informací
- *getPic* – vrací povolení či zakázání stahování obrázků
- *getArch* – vrací povolení či zakázání stahování archívů
- *getText* - vrací povolení či zakázání stahování textových souborů
- *getSound* - vrací povolení či zakázání stahování zvukových souborů
- *setDir* – nastavuje cestu pro ukládání stažených souborů
- *setNestLevel* – nastavuje úroveň zanoření
- *setCapture* – nastavuje povolení či zakázání zachytávání informací
- *setPic* - nastavuje povolení či zakázání obrázků
- *setArch* - nastavuje povolení či zakázání archívů
- *setText* - nastavuje povolení či zakázání textových souborů
- *setSound* - nastavuje povolení či zakázání zvukových souborů

#### 4.1.6 HTMLLinkExtractor

Je jednou z nejdůležitějších tříd celé aplikace. Jejím hlavním úkolem je procházení zdrojového kódu WWW stránky s postupným vyhledáváním odkazů, které následně vyhodnocuje zda jsou či nejsou podporovány. Podporované odkazy uloží do třídy *ReferenceCollection*, kterou jsme si popsali výše, a neuložené odkazy jednoduše přeskočí.

Před samotným uložením do třídy *ReferenceCollection* však ještě probíhá proces korekce tvaru URL adresy. Ta totiž může být buď v absolutním tvaru, který je pro nás vyhovující, a nebo v relativním tvaru a tento musí být do tvaru absolutního dodatečně převeden. O tento převod se automaticky stará třída *Reference*, kterou si popíšeme níže v tomto textu.

### Implementované metody :

- *extract* – vstupní metoda sloužící pro inicializaci proměnných a následné zavolání metody *extractLink*
- *extractLink* – spouští proces procházení zdrojového kódu a vyhledávání URL odkazů
- *replaceRef* – nahrazuje relativní odkaz absolutním odkazem
- *isRefTag* – zjišťuje zda se jedná o tag reprezentující odkaz
- *isScript* – zjišťuje zda se jedná o tag reprezentující script
- *isStyle* - zjišťuje zda se jedná o tag reprezentující kaskádový styl
- *isComment* - zjišťuje zda se jedná o tag reprezentující komentář
- *getNextUrl* – vrací v pořadí další nalezenou URL adresu
- *readPath* – přečte řetězec zapsaný jako hodnotu odkazu
- *skipSpaces* – přeskočí následující znaky mezery
- *skipBlock* – přeskočí nadefinovaný blok zdrojového kódu



#### 4.1.7 CSSLinkExtractor

Plní obdobnou funkci jako třída *HTMLLinkExtractor*, ale, jak už název napovídá, nejedná se o procházení HTML kódu nýbrž o procházení kaskádových stylů, které mohou být jak součástí HTML kódu, tak mohou být uloženy v samostatném souboru. Proto zahrnuje oproti třídě *HTMLLinkExtractor* některé metody navíc. Postup procházení je obdobný jako u výše zmíněného procházení a to probíhá následovně. Postupně se prochází zdrojový kód stylu dokud rozpoznávač nenarazí na klíčové slovo “URL“, které říká, že následující řetězec bude představovat URL odkaz na nějaký soubor, zpravidla se jedná o obrázek. Tento odkaz je dále vyhodnocen zda je v absolutním tvaru a pokud není je do tohoto tvaru převeden a následně uložen do fronty.

#### Implementované metody :

- *extract* - vstupní metoda sloužící pro inicializaci proměnných a následné zavolání metody
- *extractFromBlock* – vstupní metoda pro inicializování proměnných určená pro styly psané uvnitř HTML kódu
- *extractLinkFromBlock* – prochází blok kódu stylu a hledá odkazy
- *extractLink* – prochází zdrojový kód stylu a hledá odkazy
- *replaceRef* – nahrazuje relativní tvar odkazu absolutním tvarem
- *findNextUrl* – najde další URL adresu
- *readPath* – přečte celé řetězec tvořící URL adresu
- *skipSpaces* – přeskočí všechny po sobě následující znaky mezery
- *skipBlock* – přeskočí nadefinovaný blok zdrojového kódu

#### 4.1.8 Reference

Tato třída reprezentuje běžnou URL adresu, se kterou se setkáváme v odkazech na WWW stránkách. Při vytvoření instance třídy se předá konstruktoru této třídy URL adresa v libovolném tvaru a třída se už postará o vytvoření absolutního i relativního tvaru adresy. Zároveň se vytvoří cesta, ve které bude soubor umístěn při uložení do paměti zařízení a v případě ukládání stránky, u které není jednoznačně uveden název souboru se zdrojovým kódem lze tento název vygenerovat.

### Implementované metody :

- *createAbsolute* – vytvoří z relativní cesty absolutní cestu
- *createRelative* – vytvoří z absolutní cesty relativní cestu
- *generateIndexHTML* – vygeneruje soubor index.html
- *generateAnyHTML* – vygeneruje soubor stoXnis.html, kde X reprezentuje přirozené číslo, které se průběžně iteruje
- *createHierarchy* – vytvoří hierarchii adresářové cesty, ve které bude soubor uložen
- *getPathWoFile* – vrátí adresářovou cestu, ve které bude soubor uložen
- *getPathWithFile* – vrátí plnou cestu umístění souboru včetně názvu souboru
- *getFileName* – vrátí název souboru včetně přípony
- *setFileName* – nastaví název souboru
- *isAbsolute* – zjišťuje, zda je adresa absolutní
- *isFile* – zjišťuje, zda URL adresa ukazuje na soubor
- *isWebFile* – zjišťuje, zda se jedná o soubor HTML stránky
- *fileType* – zjišťuje o jaký typ souboru se jedná (stránka, styl, jiný typ souboru)
- *getAbsRef* – vrátí absolutní cestu umístění v paměti
- *getRelRef* – vrátí relativní cestu umístění v paměti

#### 4.1.9 RefDiagnostic

Jedná se o velmi jednoduchou třídu, jejíž jediným úkolem je zjistit, zda se URL adresa nachází ve stejné doméně, ve které je umístěna původně zadaná stahovaná stránka.

### Implementované metody :

- *isSupported* – jednoduché zjištění, zda je URL adresa podporována
- *isAbsoluteSupported* – zjišťuje, zda je podporována adresa v absolutním tvaru
- *isRelativeSupported* – zjišťuje, zda je podporována adresa v relativním tvaru

#### 4.1.10 TextNorm

Tato třída není pro samotný běh aplikace nutná, ale byla používána především při vývoji aplikace, kdy jsem potřeboval uspořádat zdrojový kód WWW stránek, který byl, v některých případech, značně nepřehledný. Úkolem této třídy je procházet zdrojové kódy WWW stránek a odstraňovat větší počty znaků mezery s následným nahrazením pouze jediným znakem mezery. Ovšem po zjištění překvapivého zvýšení přehlednosti jsem se rozhodl tuto třídu v aplikaci ponechat.

### Implementované metody :

- *removeUnprintableChars* – odstraní netisknutelné znaky
- *getNormalizedText* – odstraní po sobě následující znaky mezery

#### 4.1.11 Options

Třída *Options* má obdobnou funkci jako třída *Settings* tedy uchovává informace o parametrech nastavení stahování. Rozdílem je však to, že třída *Options* má tyto informace uloženy staticky a tudíž umožňuje přístup k těmto informacím ostatním třídám.

##### Implementované metody :

- *setHost* – nastaví název domény stránek
- *generateHTMLNumber* – vygeneruje přirozené číslo HTML generované HTML stránky
- *setDir* – nastavuje cestu k úložnému adresáři
- *getHost* – vrátí název domény stránek
- *getDir* – vrátí cestu k úložnému adresáři
- *setProjName* – nastavuje název projektu
- *getProjName* – vrátí název projektu

#### 4.1.12 DownloadItem

Tato třída představuje položku v seznamu stažených stránek který je zobrazen v okně Stažené. Zobrazí se vždy název projektu a informace o úspěšném či neúspěšném stažení.

Úspěšné stažení proběhne tehdy, když se během stahování program předčasně nepřerušuje a ve frontě s URL adresami tak nezůstane žádná adresa. Naopak neúspěšné stahování proběhne tehdy, když v průběhu stahování dojde k nějaké neočekávané chybě.

##### Implementované metody :

- *getProjName* – vrátí název projektu
- *setWebUrl* – nastaví URL adresu stránky
- *getWebUrl* – vrátí URL adresu stránky
- *setDownloadStatus* – nastaví příznak úspěšnosti stažení
- *getDownloadStatus* – vrátí příznak úspěšnosti stažení

#### 4.1.13 DownloadCollections

Slouží jako kolekce objektů třídy *DownloadItem*, kterou jsme si popsali výše. Položky této kolekce jsou zobrazovány v textovém okně, které se nachází na kartě stažené.

##### Implementované metody :

- *addItem* – přidá položku do kolekce
- *getNextItem* – vrátí položku z kolekce
- *isContended* – zjistí zda je položka v kolekci již obsažena

#### 4.1.14 OptionsManaging

Jedná se o třídu sloužící pro ukládání informací o nastavení aplikace, které jsou nesené třídou *Options*, do XML souboru tak aby je bylo možno při dalším spuštění aplikace opět načíst.

##### Implementované metody :

- *saveSettings* – uloží nastavení aplikace voláním metody *createSettingsFile* a následným voláním metody *saveSaved*
- *saveSaved* – metoda pro ukádání informací o nastavení aplikace do souboru
- *readDefaultSettings* – načte výchozí nastavení pro aplikaci
- *readSettings* – načte uložené informace o nastavení pro aplikaci
- *createSettingsFile* – vytvoří nový XML soubor, do kterého se budou ukládat informace o nastavení aplikace

#### 4.1.15 DataManaging

Jedná se o třídu sloužící pro ukládání informací o stažených stránkách, které jsou nesené třídou *DownloadCollections*, do XML souboru tak aby je bylo možno při dalším spuštění aplikace opět načíst.

##### Implementované metody :

- *createDownloadListFile* - vytvoří nový XML soubor, do kterého se budou ukládat informace o stažených stránkách
- *saveData* – uloží seznam stažených stránek voláním metody *createDownloadListFile* a následným voláním metody *saveDownloaded*
- *saveDownloaded* – metoda pro ukládání informací o stažených stránkách
- *loadData* – načte informace o stažených stránkách

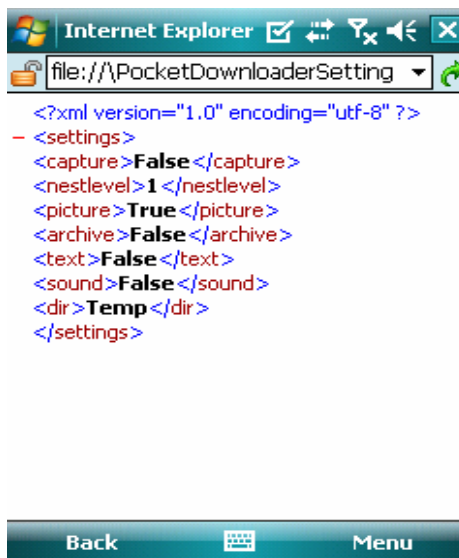
## 4.2 Ukládání informací a nastavení

Třída *DataManaging* a *OptionsManaging* slouží, jak již bylo psáno výše, k uložení informací o Stažených stránkách respektive o nastavení aplikace. Data se ukládají do XML souboru, který zajišťuje jednoduché čtení a případnou modifikaci údajů, které jsou v něm uloženy. Strukturu obou XML souborů si můžete prohlédnout na obrázcích níže.



Obrázek 8:XML soubor pro uložení stažených projektů

Ukládání stažených projektů je jednoduché a do XML souboru se název projektu zapisuje v entitě item. Na obrázku je uložen pouze jeden projekt s názvem Proj1. xml soubor pro uchovávání informací o stažených stránkách má název data.xml.



Obrázek 9 :XML soubor pro uložení nastavení

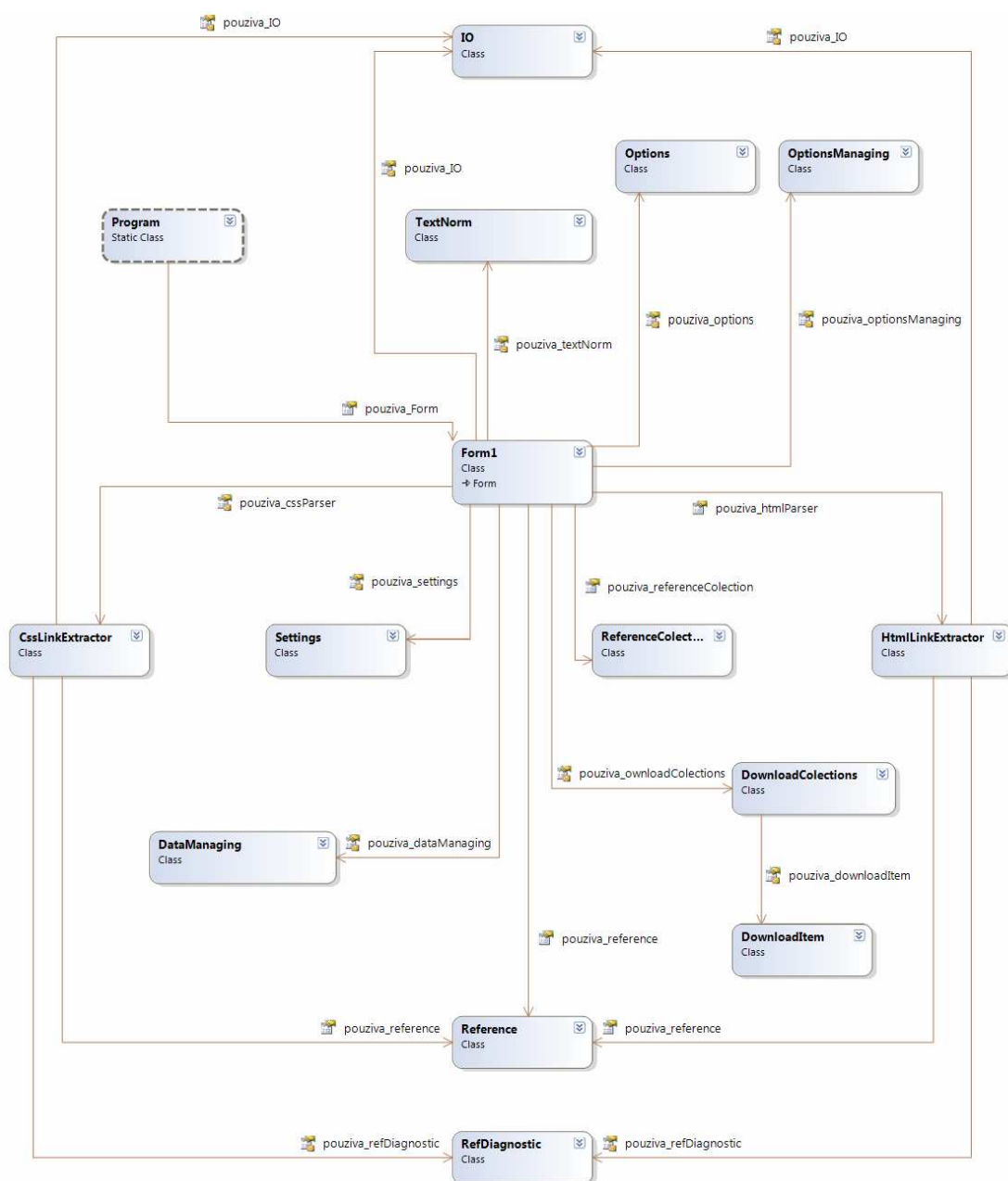
Uchovávání informací o nastavení je mírně složitější než předešlý případ. V tomto souboru jsou pevně dané entity, u kterých pouze měníme jejich hodnoty.

### **Mezi uchovávané hodnoty patří:**

- **capture** – nastavení zachytávání informací o stahování
- **nestlevel** – nastavení úrovně zanoření
- **picture** – nastavení stahování obrázků
- **archive** – nastavení stahování archívů
- **text** – nastavení stahování textových souborů
- **sound** – nastavení stahování souborů
- **dir** – nastavení cesty adresáře s ukládanými weby

### 4.3 Třídní diagram

Na obrázku níže je uveden třídní diagram znázorňující vztahy mezi jednotlivými třídami. Tento třídní diagram však, z důvodů zachování přiměřené velikosti, nezahrnuje metody a atributy jednotlivých tříd. Kompletní diagramy jednotlivých tříd jsou přiloženy v příloze k této práci.



Obrázek 10: Třídní diagram stahovače

## 4.4 Testování aplikace

Aplikaci jsem testoval na zařízení HP iPAQ 4100, které jsem dostal zapůjčeno pro účely testování vyvíjené aplikace od VŠB.

### Parametry HP iPAQ 4100

<b>Výrobce</b>	Hewlett Packard
<b>Model</b>	iPAQ 4100
<b>Operační systém</b>	Microsoft Windows Mobile 2003
<b>Rozměry</b>	113mm x 70mm x 13mm
<b>Váha</b>	132 g
<b>Procesor</b>	400 Mhz
<b>Komunikace</b>	WLAN 801.11b, Bluetooth, IrDA
<b>Paměť</b>	64MB SDRAM, 32MB Flash ROM
<b>Velikost displeje</b>	3,5 "
<b>Rozlišení displeje</b>	240 x 320 pixelů

V průběhu implementace jsem prováděl testy jak v prostředí .NET Framework, tak i následně v .NET Compact Frameworku. V prostředí .NET Framework jsem testy prováděl v konzolovém režimu za použití krokování, pomocí kterého jsem ladil parser pro vyhledávání odkazů pomocí regulárních výrazů. Obtíže mi zpočátku činily různé způsoby zápisu URL adres v HTML kódu.

Pro testování jsem využíval převážně stránek [www.jakpsatweb.cz](http://www.jakpsatweb.cz), u kterých jsem si ověřil zda jsou validní abych se vyhnul případným komplikacím při vývoji. Po odladění parseru pro tento web jsem následně provedl pro ověření testy na několika dalších webech.

### Testované weby při testech na čtení URL odkazů:

- <http://www.seznam.cz>
- <http://www.vsb.cz>
- <http://www.aukro.cz>
- <http://www.avizo.cz>
- <http://www.bonus.cz>
- <http://www.jakpsatweb.cz>

Na těchto stránkách jsem rovněž prováděl i všechny následující testy. Dalším problémem byla konverze relativních URL adres na relativní přičemž opačná konverze problémy nečinila. Bylo třeba nastudovat problematiku způsobů zápisu URL odkazů v HTML kódu. Bylo třeba zahrnout do rozpoznávání všechny tyto způsoby a umožnit tuto relativní adresu správně napojit na adresu prohlížené stránky, v jejímž zdrojovém kódu se relativní adresy nacházely.

Poslední překážkou při zpracovávání odkazů byly odkazy psané pomocí jazyka PHP a dále implicitní odkazy, které nebyly zakončené názvem souboru. V prvním případě jsem pomocí regulárních výrazů opět zkusil najít v odkazu název souboru, který bych použil a když odkaz



žádný název souboru neobsahoval, tak jsem jej nechal vygenerovat ve tvaru stoXnis.html, kde místo písmene X se nachází přirozené číslo průběžně zvyšované o číslo jedna.

V případě, že se v odkazu nenacházel ani skript a ani název souboru jsem jako název souboru přiřadil index.html. Tímto jsem měl kód pro práci s odkazy hotov a úspěšně jsem jej otestoval na všech výše uvedených WWW stránkách. Nyní již scházelo jen naimplementovat ukládání stažených stránek, což nečinilo žádné problémy.

Po zkompletování a odladění aplikace pro platformu .NET Framework jsem ji přepracoval pro prostředí .NET Compact Framework, kde jsem nemusel provádět žádné změny, jelikož všechny mnou použité knihovny byly oběma platformami podporovány. Stačilo tak pouze vytvořit uživatelské rozhraní a propojit ho s již funkčním programem.

Po odladění na platformě .NET Compact Framework jsem v emulátoru provedl stažení všech výše uvedených stránek. Z důvodů pomalé rychlosti připojení k Internetu v emulátoru jsem však používal úroveň zanoření maximálně se stupněm dva.

Nakonec jsem provedl samotné otestování aplikace na reálném zařízení.

## **Testování na PDA**

Při testování aplikace na skutečném zařízení PDA mi zpočátku činilo potíže se s zařízením seznámit a naučit se s ním pracovat. Po seznámení se s zařízením jsem narazil na problém s připojením k Internetu prostřednictvím domácí WiFi sítě. Tento problém se mi nepodařilo vyřešit a proto jsem přistoupil k možnosti připojení skrze program ActiveSync, kdy jsem PDA připojil kabelem k počítači, na kterém jsem měl připojení k Internetu již k dispozici. Poté jsem provedl samotné testy na výše zmíněném seznamu WWW stránek. Při testování aplikace na kapesním počítači jsem však narazil na problém, kdy se některé stránky stahují delší dobu a v některých případech aplikace během stahování přestane reagovat a je nutné ji vypnout. Tento problém se mi dosud nepodařilo vyřešit ani zjistit jeho přesnou příčinu. V úvahu však připadají dvě možnosti. První možností je ta, že PDA nemá dostatek paměťových prostředků a aplikace není zcela optimalizována. Druhou možností by pak bylo pomalé připojení k internetu na zařízení PDA.

## 5. Závěr

Při vypracovávání této práce jsem využil své znalosti získané při studiu na VŠB a prohloubil jsem své znalosti z oboru tvorby WWW stránek a programování v jazyku C#.

Pro psaní zdrojových kódů jsem použil vývojové prostředí Microsoft Visual Studio 2008 Professional Edition.

V první fázi vývoje jsem vytvořil verzi pro použití na stolním počítači s operačním systémem Windows 7. V této fázi vývoje jsem aplikaci ladil a snažil se ošetřit všechny výjimky. Po úspěšném provozu na stolním počítači jsem teprve začal aplikaci implementovat pro kapesní počítač. Během vývoje aplikace mi největší potíže činil parser, který URL adresy vyhledává a konvertor relativních adres na adresy absolutní. Po vyladění těchto částí již žádné problémy nenastaly. Výsledná aplikace je odladěna proti chybnému zadání vstupních parametrů a byla odzkoušena jak na statických tak na dynamických WWW stránkách. Testy však proběhly úspěšně pouze v emulátoru. Při běhu aplikace na reálném PDA se během stahování stává, že aplikace přestane reagovat a je nutné ji vypnout. Na tomto problému stále pracuji a řeším její optimalizací celého zdrojového kódu pro zajištění co nejmenších paměťových nároků. Aplikace bohužel nepodporuje odkazy v jazyce Javascript a tudíž se stránky odkazované těmito odkazy nestahují.

Aplikace je psána tak, aby byla do budoucna dále rozšiřitelná a proto je možno podporu pro Javascript dodatečně přidat. Dalším nedostatkem by poté bylo rozhraní aplikace, které je velmi jednoduché a nepraktické. Do budoucna by tedy bylo možno aplikaci optimalizovat tak, aby požadovala menší nároky na výpočetní výkon a paměť PDA. Dále by bylo možno aplikaci aktualizovat o nový vzhled, přidat podporu Javascriptu, umožnit stahovat pouze soubory bez WWW stránek a zdokonalit práci s adresáři při ukládání WWW stránek.

## Literatura

- [1] <http://www.wikipedia.org>, Wikipedie, otevřena encyklopedie.
- [2] <http://digitalne.centrum.cz>: Jak stáhnout web a surfovat offline, 22.03.2009
- [3] <http://technet.idnes.cz>: Surfujte na Internetu bez připojení
- [4] <http://www.stahuj.centrum.cz>: HTTrack Website Copier, WebSite extraktor, Teleport Pro, Offline explorer

## **A. CD-ROM**

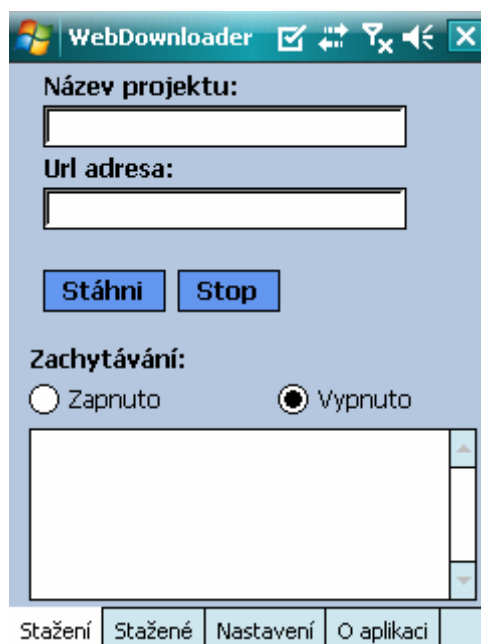
### Obsah CD ROM

- 1) Text bakalářské práce ve formátu DOC a PDF
- 2) Uživatelská příručka aplikace
- 3) Zdrojové kódy a zkompilevané třídy aplikace
- 4) Ukázky stažených stránek

## B. Uživatelská příručka:

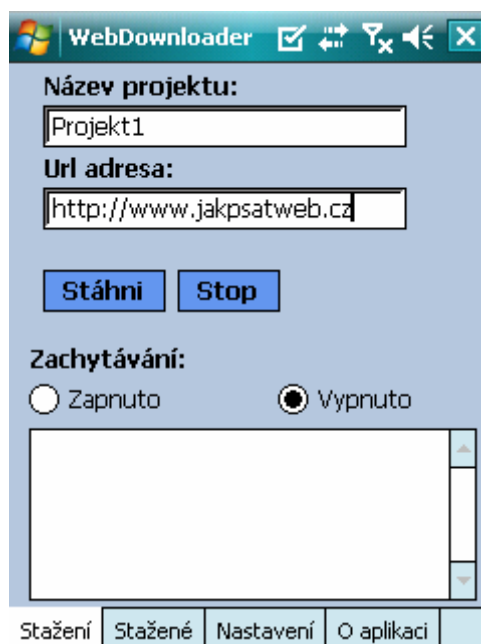
### ***Postup při stahování www stránek***

Pomocí názorného příkladu si ukážeme postup jak stáhnout požadované stránky. Ze všeho nejdříve se ujistíme, zda máme funkční připojení k internetu. Nyní si již můžeme spustit samotnou aplikaci a mělo by se nám zobrazit následující okno se zobrazenou kartou Stažení.

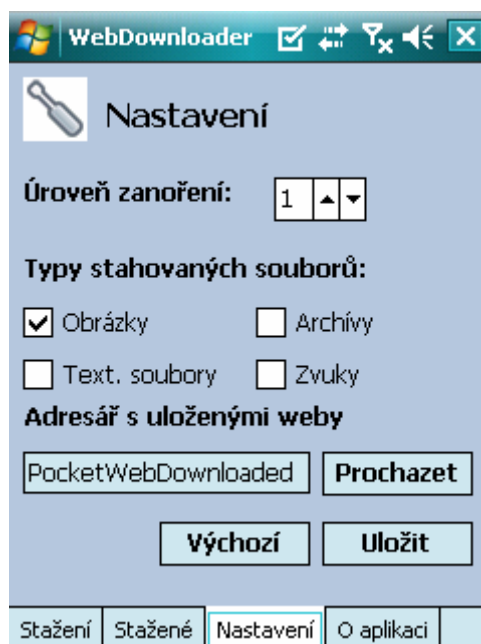


Zde můžeme vidět textová pole pro zadání názvu projektu, URL adresy, tlačítka pro spuštění a přerušení stahování a nakonec textové pole pro zobrazování informací o průběhu stahování, které můžeme pomocí přepínacích políček zapnout či vypnout. Jak název projektu, tak i URL adresa jsou parametry, které jsou povinné tudíž je nutné je pro započetí stahování zadat. Zadání údajů se provede tak, že klikneme stylusem na příslušné textové pole a pomocí zobrazené klávesnice zadáme příslušný řetězec znaků. V případě URL adresy zadáváme adresu i se schématem HTTP nebo HTTPS.

Obrazovku se zadanými údaji si můžete prohlédnout níže.

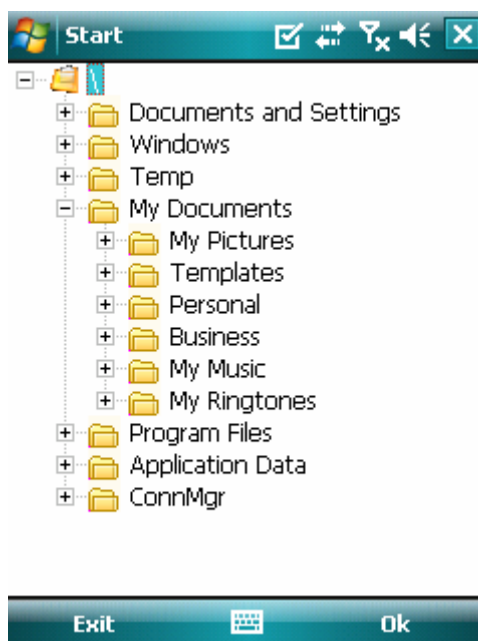


Následně si přepneme na kartu Nastavení viz. obrázek níže.

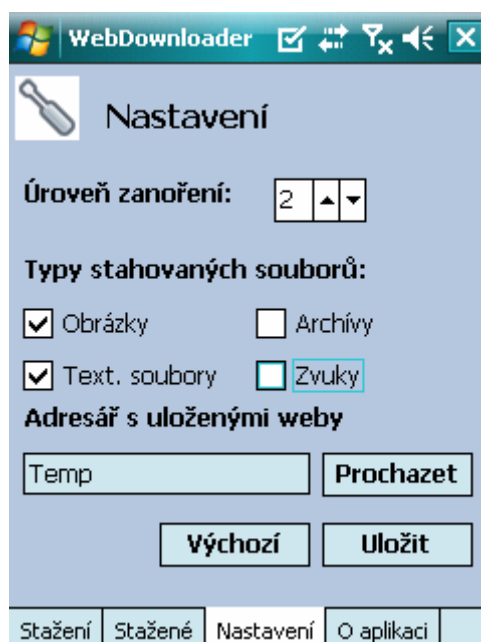


Obrazovka, kterou vidíte zobrazuje výchozí nastavení. Pomocí parametru úroveň zanoření nastavíme hloubku stahování www stránek, kterou můžeme nastavit v rozmezí 0 – 10, kdy při zadání parametru 0 stáhneme pouze zadanou www stránku. Pomocí zaškrtnutých políček navolíme typy souborů, které se budou spolu se stránkami stahovat. Posledním možným nastavením na této kartě by bylo zadání cesty k adresáři pro stahované stránky.

Při kliknutí na tlačítko procházet se nám zobrazí hierarchie adresářů.



Klikáním na jednotlivé adresáře si navolíme požadovanou cestu pro ukládání stahovaných WWW stránek. Po navolení cesty klikneme na tlačítko Ok čímž navolenou cestu potvrdíme a tato cesta se nám vypíše do příslušného textového pole na Kartě Nastavení. Na obrázku níže můžete vidět jednu z možných variant nastavení.



Při čtení údajů z karty můžeme vidět, že je aplikace nastavena pro procházení WWW stránek do úrovně zanoření stupně dva, dále že se spolu se stránkami budou také stahovat obrázkové a textové soubory a všechny stahované soubory se budou ukládat do adresáře Temp.

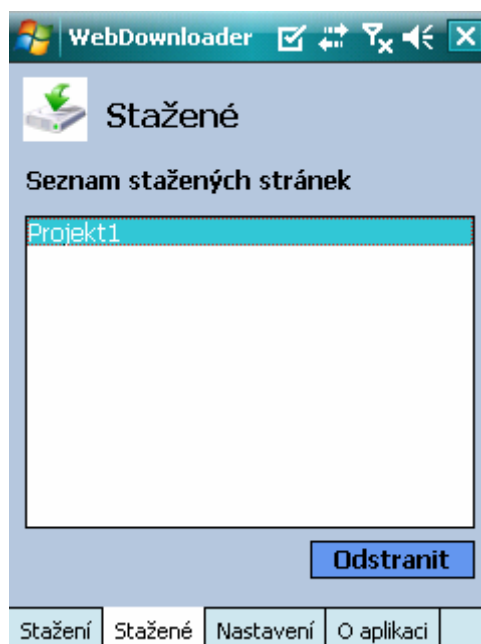
Kliknutím na tlačítko výchozí by se nastavení vrátilo do svého výchozího stavu. Tlačítkem uložit naše nastavení uložíme do XML souboru, ze kterého se toto nastavení při dalším spuštění aplikace opětovně načte. Výpis kódu XML souboru odpovídajícího nastavení na předchozím obrázku si můžete prohlédnout na obrázku níže.



Nyní máme veškeré potřebné údaje zadány a můžeme naši stránku směle stahovat. Stahování zahájíme kliknutím na tlačítko Stáhni na kartě Stažení. Po stisknutí se nám tlačítko deaktivuje, což symbolizuje, že probíhá stahování. Stahování bude dokončeno po opětovném aktivování tlačítka. Celý průběh stahování můžeme také sledovat v textovém poli na kartě Stažení.

Po úspěšném stažení stránky se nám v textovém poli na kartě Stažené zobrazí název projektu staženého webu. Nyní když se podíváme do naší složky, do které stahujeme stránky, tak tam uvidíme novou složku nazvanou názvem projektu, který jsme zadali při stahování. V této složce se již nachází naše požadovaná, kompletně stažená stránka.





Do tohoto seznamu se nám postupně budou přidávat další názvy projektů. Jednotlivé projekty můžeme odstranit kliknutím na název projektu a následným kliknutím na tlačítko odstranit.

Poslední kartou by byla karta O aplikaci, která má pouze informativní charakter a slouží k stručnému popisu funkce aplikace a k sdělení údajů o autorovi.

